DU Journal of Undergraduate Research and Innovation

Volume 1, Issue 3 pp 211-218, 2015

Extension of the Josephus Problem with Varying Elimination Steps

Shivam Sharma, Raghavendra Tripathi, Shobha Bagai, Rajat Saini, Natasha Sharma
shobhabagai@gmail.com, Cluster Innovation Center, University of Delhi, Rugby Sevens Building, University Stadium, G.C. Narang-Marg, New Delhi, India

ABSTRACT

This paper is an effort of expounding the recursive formula for the Josephus problem (which is the essence of the popular game "Akkad Bakad Bambai Bo" played in Indian households). The academic extension of this game has been to deal with the cases where elimination steps vary with each turn. Josephus problem is essentially heralded with objective of finding the position of the player who survives the game. Most of the available literature deals with the problem of finding the survival's position when sequential elimination of persons takes place in k steps for n circled people. There is a simple algorithm known to solve this problem; however the rigorous procedure can be replaced with a formula. The nature of this formula can be either recursive or non-recursive; both of which comprise this paper. We have extended the problem to the situation wherein the elimination step k varies with iterations.

The non-recursive methods include the binary method & paper [4] contains non-recursive formula for constant elimination step $k = 2$ and the recursive formula has been derived using dynamic recursive methods. This dynamic recursive formula can be modified to withstand different extensions of Josephus problem; making it all-inclusive. This work shows the modifications made in the dynamic recursive formula to predict the survival's positions for varying elimination step k. We have explained the dynamic recursive formula for steps k changing as the count does (1,2,3,4 ...) and steps changing as multiples of a number ($u \in N$, $k = u, 2u, 3u, ...$). A simulation game for various facets of the Josephus problem have also been programmed using the software "*MATLAB*" which provides options to select various step sequences and to view the actual elimination process in the output table.

Keywords: Dynamic recursive method, Extended Josephus problem, Josephus Problem, Josephus Problem MATLAB Game and Akkad Bakad Bambai Bo, Recursive Formula.

INTRODUCTION

The Josephus problem has been named after Flavius Josephus, who with other 40 members was blocked by Romans in a cave and they all decided to kill themselves by forming a circle and killing themselves in step of three over capture by the Romans. Josephus states that by luck or possibly by the hand of God, he and another man remained in the end and gave up to the Romans.

So, effectively the problem is that a group of n people arranged in a circular format are numbered from 1 to n sequentially. We start eliminating every second person. For example, if $n = 6$, the people are removed in the order 2, 4, 6, 3, 1, and the last person to survive the game is intuitively the one standing at the position number 5. Let $T(n)$ denote the last person remaining when we started with n people. We consider our objective to find a simple way to compute $T(n)$ for any positive integer $n \geq 1$ when step k changes as the count $(1,2,3,4 \ldots)$ and steps change as multiples of a number ($u \in N,\ k = u, 2u, 3u, \ldots$) where $k > 0$.

The literature by Miguel Lerma's proof of the Josephus problem gives the relationship between powers of two and binary numbers to calculate winning position in which a non–recursive binary method has been depicted in order to calculate the survival position as mentioned in the methodology section. Using induction it has been proved that regardless of the number of participants, person 1 survives as long as the number of person are in powers of 2. As long as the number of people in the circle is even in each pass the first person remains untouched. Paper [4] proposes and proves non-recursive and recursive formulae for elimination step $k = 2$ only. R. Graham and D. Knuth [1] have shown an iterative formula for calculating the next survival position. The dynamic recursive method as explained in the next section can be extended to withstand with variable elimination steps (k) . This paper explains the extension of the dynamic recursive method to calculate the survival position with elimination steps changing as sum sequence and multiple of a number.

METHODOLOGY

Explanation for recursive formula for step $k = 2$

k – represents the eliminating step
n – represents total number of people
$T(i)$ – represents the survival position when *i* is the total number of people
Recursive formula for constant step $k = 2$ and n people:

The position of the survival in $T(2n)$ when every second person out of n people standing in a circle is eliminated until only one survives is given by the recurrence relation
$T(2n) = 2T(n) - 1\ ;\ T(1) = 1.$

Explanation: Let *i* be the position of any survivor's position after first elimination. Clearly, the corresponding position in the circle having $2n$ elements is $2i - 1$. Hence, $T(2n)\ = 2T(n) - 1$.

n	*survival position in n people is i*	I	$2n$	*Survival's position in* $2n$ *is* $2i - 1$
1	1	1	2	1
2	1 2	1	4	1
3	1 2 3	3	6	5
4	1 2 3 4	1	8	1
5	1 2 3 4 5	3	10	5
6	1 2 3 4 5 6	5	12	9
7	1 2 3 4 5 6 7	7	14	13

Table-I: Table showing the survival's position in $T(2n)$

The position of the survival in $T(2n+1)$ when every second person out of n persons standing in a circle is eliminated until only one survives is given by the recurrence relation
$T(2n) = 2T(n) + 1$; $T(1) = 1$

Explanation: Let i be the position of any survivor element after first elimination. Clearly, the corresponding position in the first circle having $2n + 1$ elements is$2i + 1$. Hence, $T(2n + 1) = 2T(n) + 1$.

n	*survival position in n is i*	i	$2n + 1$	*Survival's position* $2i + 1$
1	1	1	3	3
2	1 2	1	5	3
3	1 2 3	3	7	7
4	1 2 3 4	1	9	3
5	1 2 3 4 5	3	11	7
6	1 2 3 4 5 6	5	13	11
7	1 2 3 4 5 6 7	7	15	15

Table-II: Table showing the survival's position for $T(2n + 1)$

Non-Recursive binary method for step $k = 2$:

Another way of finding the survivor's position $T(n)$ for n persons with constant step $k = 2$ is binary method [6]:

Therefore,$T(n)$ = left rotation of the binary digits of n
Suppose for n (base 10) =$x_1x_2x_3 \ldots x_n$(base 2)where x_k is the binary values of n with $x_1 = 1$. Then,
$T(n) = x_2x_3 \ldots x_nx_1$

For example, for $n = 6$ with $k = 2$
$(6)_{10} = (110)_2$ since x_3 in binary expansion of $(6)_{10}$ is 1
So, the survival's position is $T(6) = (101)_2 = (5)_{10}$

Similar kinds of recursive formulas exits for step $(k) = rr \epsilon N$

Dynamic recursive formulae:

So, another way of computing the survivor's position $T(n)$is by using the dynamic recursive method [5] [7] which have $O(n)$ running time complexity.
Before using the formula let us fix some notations mentioned in [5] for the recursive formula.

Number the n positions in the circle by $1, 2, 3 \ldots n$ and start counting at number 1. Then every k^{th} element is removed. It is then defined as

$$T(n, k, i), \quad (n > 1;\ k > 1;\ 1 < i < n)$$

to be the number of the i^{th} element which is removed by the process described above (see example in Figure-I).

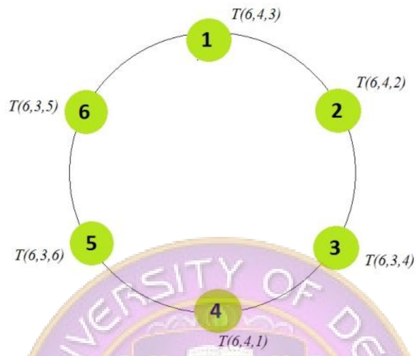

Figure-I.

Here the $T(n, k, i)$ denotes the person eliminated at i^{th}step.

In [5] the recursive formula for the $T(n, k, i)$ has been explained and it states that for Josephus problem the following recursion holds:

$$T(n, k, i) = (T(n-1, k, i+1) + k - 1) mod(n) + 1, \quad (n > 1;\ k > 1;\ 1 < i < n)$$

$$\text{With initial value } T(1, k, i) = 1 \text{ or } T(n, k, 1) = (k-1) mod(n) + 1$$

It is proved in [5]

"Suppose, we know the value of $T(\square, k, i) = g$. Hence, if we start counting at number 1, the i^{th} member removed is number g. Now consider $T(n, k, i + 1)$. Because of $T(n, k, 1) = (k - 1) mod(n)$, in the first step number $(k-1) mod(n) + 1$ is removed and for the second step we start counting at number k+1. Therefore the problem is to find the i^{th} member which is removed (after removing $(k - 1)\ mod\ (n) + 1)$ when we start counting at number$k + 1$. But this is $(g + k - 1)\ mod\ (n)$. So, we get the recursive formula."

If we know that $T(n, k, i) = g (g \in N)$, then we start counting at 1 then the i^{th} member removed is g.
Now we need to find the number removed in the next step when we start counting from k+1 number i.e. we need to find $T(m, k, i + 1)$

So, the $T(m, k, i + 1) = (g + k - 1)\%n + 1$

For example, the survival position for $n = 11$ with constant step $k = 3$ is calculated and the output calculated recursively is shown in the Table-III.

No of People left (n)	Elements Remained in the sequence	Survival's Position $f(n,k) = (f(n-1,k) + k - 1)\%n + 1$ $f(1,k) = 1 \text{ with initial } k = 3$
11	**1 2 3 4 5 6 7 8 9 10 11**	**f(11,3) = (f(10,3) + 2)%11+1= 7**
10	**1 2 3 4 5 6 7 8 9 10**	f(10,3)=(f(9,3)+2)%10 +1= **4**
9	**1 2 3 4 5 6 7 8 9**	f(9,3)=(f(8,3)+2)%9 +1=**1**
8	**1 2 3 4 5 6 7 8**	f(8,3)=(f(7,3)+2)%8 +1=**7**
7	**1 2 3 4 5 6 7**	f(7,3)=(f(6,3)+2)%7 +1= **4**
6	**1 2 3 4 5 6**	f(6,3)=(f(5,3)+2)%6 +1 = **1**
5	**1 2 3 4 5**	f(5,3)=(f(4,3)+2)%5 +1 = **4**
4	**1 2 3 4**	f(4,3)=(f(3,3)+2)%4 +1 = **1**
3	**1 2 3**	f(3,3)=(f(2,3)+2)%3 +1 = **2**
2	**1 2**	f(2,3)=(f(1,3)+2)%2 +1 = **2**
1	**1**	f(1,3) = **1**

Table-III: Table showing the survival's position for constant $k = 3$ elimination step using the dynamic programming method

RESULTS

Mathematical Analysis

Extended Josephus Problem:
Extended Josephus problem is to determine the survivor's position $T(n)$ with variable elimination steps i.e., we assume that, unlike the original Josephus problem where second person was being eliminated each time, at every count "being-eliminated number" is different. For instance we eliminate second person in 1^{st} iteration, then 2^{nd} in second iteration and so on. Survivor's position can easily be calculated with slight modification in the dynamic recursive method.

Steps changing as counting:
For n people with steps changing as a counting sequence
For instance k takes the sequences $2, 3, 4, 5, 6$ ... after elimination of person at each step with initial step $k = 2$
So, the dynamic recursive formula becomes

$$f(n,k) = (f(n-1, k+1, i+1) + k - 1)\%n + 1$$

For example, the survival's position for $n = 11$ people with initial step $k = 3$ is calculated using the dynamic programming method and the output calculated iteratively is shown in the Table-IV.

No of People left (n)	Element Remained	Survival's Position *f(n,k)=(f(n-1,k+1,i+1)+k-1)%n +1* *f(1,k)* = 1 with initial *k=3* , *i=1*
➡**11**	**1 2 3 4 5 6 7 8 9 10 11**	**f(11,3,1) = (f(10,4,2) + 2)%11 +1=10**
10	**1 2 3 4 5 6 7 8 9 10**	f(10,4,2)=(f(10,4,2) + 2)%11 +1=**7**
9	**1 2 3 4 5 6 7 8 9**	f(9,5,3)=(f(10,4,2) + 2)%11 +1=**3**
8	**1 2 3 4 5 6 7 8**	f(8,6,4)=(f(10,4,2) + 2)%11 +1=**7**
7	**1 2 3 4 5 6 7**	f(7,7,5)=(f(10,4,2) + 2)%11 +1=**1**
6	**1 2 3 4 5 6**	f(6,8,6)=(f(10,4,2) + 2)%11 +1=**1**
5	**1 2 3 4 5**	f(5,9,7)=(f(10,4,2) + 2)%11 +1=**5**
4	**1 2 3 4**	f(4,10,8)=(f(10,4,2) + 2)%11 +1=**1**
3	**1 2 3**	f(3,11,9)=(f(10,4,2) + 2)%11 +1=**3**
2	**1 2**	f(2,12,10)=(f(10,4,2) + 2)%11 +1=**1**
1	**1**	**1**

Table-IV: Table showing the survival's position with steps as sequence and initial $k = 3$ elimination step using the dynamic programming method

Steps changing as multiples:
For instance k takes the values with initial step $k = 2$ as $2, 4, 6, 8$ ...
With initial step $k = 3$, it takes the values $3, 6, 9$ ...
So, the dynamic recursive formula becomes

$$f(n,k) = (f(n-1,k,i+1) + k*i-1)\%n + 1 \quad where\ 0 < i < n$$

For example, the survival's position for $n = 11$ people with initial step $k = 3$ is calculated using the dynamic recursive method and the output calculated iteratively is shown in the Table-V.

No of People left (n)	Element Remained	Survival's Position $f(n,k) = (f(n-1,k,i+1) + k*i-1)\%n + 1$ $f(1,k) = 1$ with initial $k = 3$
➡**11**	**1 2 3 4 5 6 7 8 9 10 11**	**f(11,3,1) = (f(10,3,2) + 2)%11+1= 6**
10	**1 2 3 4 5 6 7 8 9 10**	f(10,3,2)=(f(9,3,3)+ 5)%10 +1= **3**
9	**1 2 3 4 5 6 7 8 9**	f(9,3,3)=(f(8,3,4)+ 8)%9 +1= **7**
8	**1 2 3 4 5 6 7 8**	f(8,3,4)=(f(7,3,5)+ 11)%8 +1= **7**
7	**1 2 3 4 5 6 7**	f(7,3,5)=(f(6,3,6)+ 14)%7 +1= **3**
6	**1 2 3 4 5 6**	f(6,3,6)=(f(5,3,7)+ 17)%6 +1 = **2**
5	**1 2 3 4 5**	f(5,3,7)=(f(4,3,8)+ 20)%5 +1 = **2**
4	**1 2 3 4**	f(4,3,8)=(f(3,3,9)+ 23)%4 +1 = **1**
3	**1 2 3**	f(3,3,9)=(f(2,3,10)+ 26)%3 +1 = **1**
2	**1 2**	f(2,3,10)=(f(1,3,11)+ 29)%2 +1 = **1**
1	**1**	1

Table-V: Table showing the survival's position with multiples of the initial *k = 3* elimination step using the dynamic programming method

Implementation of the Algorithm

A game based on the Josephus Algorithm has been built to predict the survival's position and a GUI to depict the Josephus game simulation

Figure-II.GUI layout of the game

The game has two options - either a person can play the simulation or he can predict the survival position and the graphical user interface would give him the correct answer referring to the different options of eliminating steps.

Playing the simulation for the Josephus Problem:

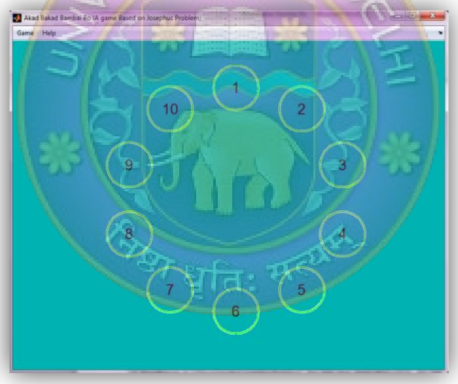

Figure-III. The simulation layout

The outputs after elimination; the positions of elimination have been marked with red circles and the green circle shows the survivor in the Figure-IV.

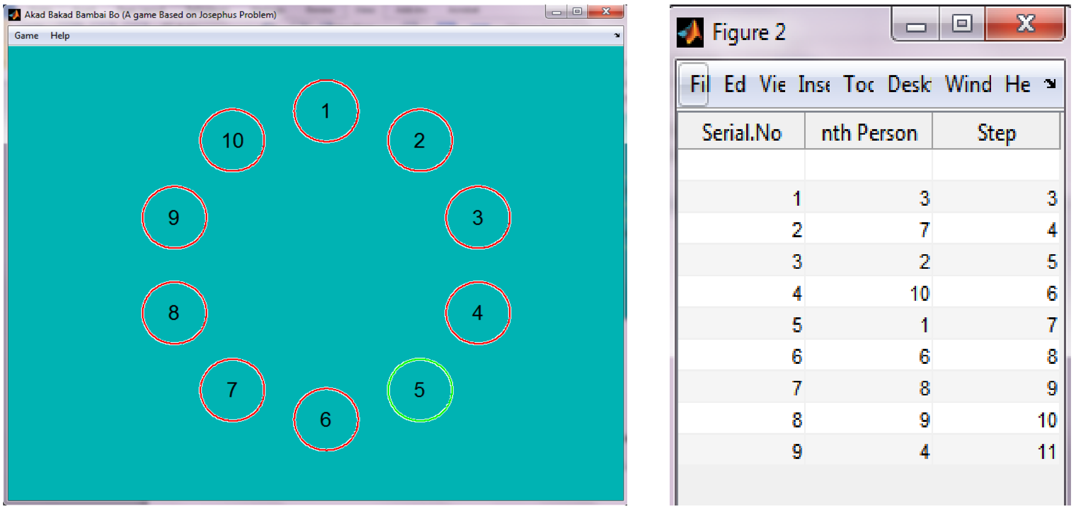

Figure-IV. Figure after elimination of all persons and with only one survived person and with the output table on the right side depicting the person eliminated at each step

CONCLUSIONS

We have successfully implemented the Josephus's problem mathematically in MATLAB by first keeping k constant and then taking recursive values of k to make the game more interesting. Various methods and formulae for extended Josephus problem have been explained. The implementation is based on the recursive formula so going beyond a certain number of recursion requires permission in MATLAB for $n > 500$.

ACKNOWLEDGEMENT

We would like to acknowledge Cluster Innovation Center (CIC), DU for providing computational resources and an active environment for carrying out such work.

REFERENCES

1. R. Graham, D. Knuth, O. Patashnik Concrete Mathematics, 2nd edition, Addison-Wesley, 1994
2. RobinsonW.J The Josephus Problem Mathematical Gazette 44(1960), 47-52
3. Woodhouse, D.
   The Extended Josephus Problem, Rev. Mat. Hisp. Amer (4)33 (1973), 207-218
4. Armin Shams-Baragh Formulating the Extended Josephus Problem (December 2002)
5. Lorez Halbeisen, ETH Zurich Norbert Hunger Buhler, University of Freiburg The Josephus Problem (1997)
6. Miguel A. Lerma Josephus Problem (2003)
7. Wikipedia –Josephus Problem, Link http://en.wikipedia.org/wiki/Josephus_problem